OpenFOAM on HPC Systems

- What is OpenFOAM®?
- License Terms and Usage Conditions
- Getting Started
  - Pre- and Post-Processing
  - Batch Jobs on LRZ Clusters
  - GPFS parallel Filesystem on the LRZ HPC clusters!
- Getting Help / Documentation
- Legacy Versions

What is OpenFOAM®?

In its simplest installed form, OpenFOAM® is a set of libraries and solvers for the solution of a wide range of problems in Fluid Dynamics and Continuum Mechanics from laminar simple regimes to DNS or LES including reactive turbulent flows. Many solvers have multiphysics features, where structural mechanics domains are coupled with CFD domains or new feature available is a module for Molecular Dynamics (MD).

But OpenFOAM® also represents a C++ framework for manipulating fields and solving general partial differential equations by means of finite volume methods (FVM) on unstructured grids. Thus, it is easy to adapt to complex geometries and wide spectra of configurations and applications. Furthermore, this software is MPI parallel, and it has a large set of useful utilities that allow importing and exporting meshes, configurations, etc., and they interface to, amongst others, Fluent, CFX, Paraview, and Ensight.

License Terms and Usage Conditions

OpenFOAM® is published under a Common Creative License (ESI), and source code is freely available. Users are encouraged to download and compile a preferred version/flavor of OpenFOAM on the LRZ HPC cluster.

ESI OpenFOAM, The OpenFOAM Foundation, Foam Extend

The installation procedure is a bit clumsy - although one can in general follow the installation instructions guide of the respective OpenFOAM distributors. Decisions you have to make are about the MPI-flavor (Intel MPI, IBM MPI, OpenMPI, etc.), and the usage of the various Third-Party libraries. The best recommendation is usually to compile any dependency you need by yourself. Then namely you can be sure to be independent (to a large extent) of the LRZ system.

In cases you need help, please contact the LRZ ServiceDesk.

We offer some support for maintained OpenFOAM installations (see next paragraph). The decision on version and flavor is guided by the size of request groups. We can support only fully released OpenFOAM versions (no development versions).

Getting Started

Check the available (i.e. installed) versions:

```bash
$ module avail openfoam
------------------ /lrz/sys/share/modules/files/applications -------------------
onopenfoam/v1706+  openfoam/v1806+(default)  openfoam/v1806+_knl
----------- /lrz/sys/spack/18.2/modules/x86_avx2/linux-sles12-x86_64 -----------
onopenfoam-com/1806-impi-i32  openfoam-com/1806-impi-i64
```

More and more, we recommend to use the Spack modules.

```bash
$ module load openfoam-com/1806-impi-i32
```

As a first step, you might consider copying the large suite of OpenFOAM tutorials into your FOAM_RUN directory by invoking:

```bash
$ mkdir -p $FOAM_RUN
$ cp -r $FOAM_TUTORIALS $FOAM_RUN/
```

Smaller serial tutorial cases can be run on the login node. The larger tutorial cases, especially the MPI parallel cases, must be submitted to the HPC clusters (see below).

Pre- and Post-Processing
For pre- and post-processing, i.e. meshing or visualizing results, the LRZ Remote Visualization System is a possible option. Paraview is the visualization tool for OpenFOAM.

For post-processing using Paraview, you only need to create a file with ending `.foam` (e.g. `touch bla.foam`), and open that file from ParaView.

You can either download your data to your PC/Laptop and analyze them there. Or, for larger cases, you can use one of the options ParaView offers to analyze the data in place (i.e. remotely on the LRZ systems). In order that this works e.g. in a parallel fashion, you need to leave the case decomposed, and start `paraview` with exactly as many MPI tasks as processor folders are there. Alternatively, you can use `reconstructPar` and `decomposePar` to change the composition. In the GUI, you need to specify (hook in the right place) that you wish to open the decomposed case. If the number of MPI tasks does not match the number of processor folders, you will earn error messages.

**Batch Jobs on LRZ Clusters**

Production runs and longer simulations must be performed on the HPC clusters. A SLURM job on the Linux cluster looks for example like:

```bash
#!/bin/bash
#SBATCH -o /directory/file_output.out
#SBATCH -D /my/initial/directory/
#SBATCH -J my_job_name
#SBATCH --clusters=mpp2
#SBATCH --nodes=2  # 2 Haswell nodes on CoolMUC2
#SBATCH --tasks-per-node=28  # for 28 physical cores on CoolMUC-2
#SBATCH --export=NONE
#SBATCH --get-user-env
#SBATCH --time=02:00:00
/etc/profile
/etc/profile.d/modules.sh
module load openfoam-com/1806-impi-i32
mpiexec interFoam -parallel
```

For different LRZ systems, please consult the documentation for further or different Slurm and Batch job settings! Submission is done via `sbatch myfoam.sh`.

In order that this works correctly, the total number of MPI tasks (nodes x tasks-per-node) must be equal to numberOfSubdomains inside system /decomposeParDict.

**GPFS parallel Filesystem on the LRZ HPC clusters!**

OpenFOAM produces lots of small files - for each processor, every writeout step, and for each field. GPFS (i.e. WORK and SCRATCH at the LRZ) is not made for such a situation, and suffers. And OpenFOAM does not seem to support any HDF5 or NetCDF output, currently, to face this and similar issues. So, please users, think about the your working problem before running it brute-force on the HPC clusters with possibly hundreds or even thousands of parallel threads! What do you want to get out of your simulation? What are the relevant questions to be answered?

**Did you know?** OpenFOAM offers some features for online post-processing. This mitigates the problem with GPFS, because it reduces the necessity of storing hundreds of thousands of small files for the offline post-processing using e.g. Paraview.

In the following, there are some links of good practice:

- GPFS and OpenFOAM: Good I/O Practice Guide
- OF User Guide: Function Objects

**Getting Help / Documentation**

Tutorials and guides can be found in the official OpenFOAM Documentation.

**Legacy Versions**

Old versions of OpenFOAM cannot be supported on LRZ-systems (OF-versions < 4.x). You need to port to one of the supported recent versions of OpenFOAM. (OF-versions >= 4.x) Specifically, the older versions do not support the vector-register hardware of modern CPUs! And generally, we recommend to use the latest stable release available.